Flash questions in ILIAS Test & Assessment

Helmut Schottmüller

Flash questions in ILIAS Test & Assessment

Helmut Schottmüller

Published January 2009

Table of Contents

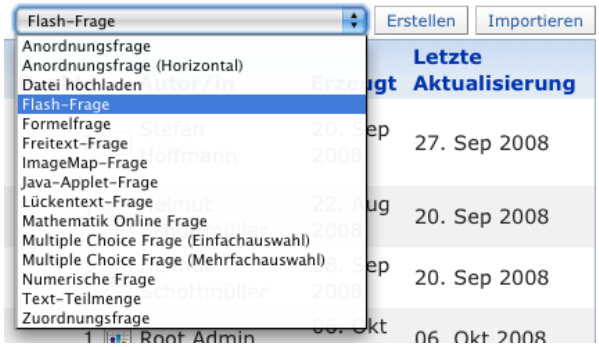
Chapter 1. Using flash questions in ILIAS Test & Assessment

Description of the question type

The flash question type in ILIAS adds the possibility to embedd Adobe Flash applets into an ILIAS question, whereas the flash applet offers the question and the logic of the question to communicate with the ILIAS system. The results will be trasferred via the ILIAS SOAP interface and will be used for the grading of a test participant.

There are no limits in the decoration and definition of a question. Only the ILIAS SOAP functions and the predefined parameters have to be considered.

A new flash question can be added into an ILIAS questionpool by selecting the flash question type in the list of questions an pressing the create button.



Adding a flash question to a question pool

Editing a flash question

After creating a flash question in ILIAS you can edit the question in a question editor. Besides the usual settings like question title, description, points etc., there is the possibility to upload a flash applet into the question. Please click on Browse in the **Flash file** section to upload a flash applet from your computer.

You can enter the flash parameters width and height for the size of the presentation. A preview will be shown in the editor as well.

Using the New parameter button you can add a key/value parameter pair that will be added to the parameter list of the flash applet during the execution in the test. Existing parameters can be removed by checking the *Delete parameter* checkmark besides the parameter.

If you need to delete the flash applet you can either upload another applet and replace the existing applet or you can check the checkbox *Delete existing file* and save the question.

Editing a flash applet in the question editor

Embedding the flash applet into the ILIAS code

Here you see how ILIAS embedds flash applets into the ILIAS code when a question is created inside a test. The embedding code is created automatically by ILIAS, it is just to give you an idea how the flash question works.

```
<object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
 codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=6,0,40,0"
 width="400" height="200" id="movie_904820">
 <param name="movie" value="frog.swf">
 <param name="FlashVars" value="speed=7&tonguelength=15">
 <embed src="http://PATH_TO_ILIAS/data/ilias3/assessment/18981/904820/flash/frog.swf" quality="high"
  bgcolor="#FFFFFF" width="400" height="200" name="movie_name_904820"
  type="application/x-shockwave-flash"
  pluginspage="http://www.macromedia.com/go/getflashplayer" FlashVars="speed=7&tonguelength=15">
 </embed>
</object>
```

Chapter 2. Developer documentation

To enable the flash applets to communicate with ILIAS, you have to provide the following preconditions:

Parameter passing from ILIAS to the flash applet

ILIAS passes some important parameters to the flash applet that has to be considered by the applet to create a valid response for the ILIAS system.

The parameters will be passed in the HTML parameter **FlashVars**. The parameters are:

- **session_id**: Session-ID of the actual test participant.
- **client**: Name of the ILIAS client
- **points_max**: The maximum number of available points for the question. The applet should always calculate points between 0 and this number for the returned points.
- **server**: URL of the ILIAS SOAP server
- **pass**: Number of actual test pass (0, 1, 2, ...)
- **active_id**: ID of the actual test participant
- **question_id**: ID of the test question

Communication between the flash applet and ILIAS using ILIAS SOAP Webservices

To communicate with ILIAS, the flash applet has to implement a SOAP client for further communication. The necessary parameters to call the SOAP functions are passed by ILIAS to the applet during the runtime using the FlashVars parameter (see above). The implementation of the SOAP client has to be done inside the flash applet.

The following SOAP functions are available for Test & Assessment:

```
saveQuestion(sid, active_id, question_id, pass, solution)
```

Parameter

- sid: xsd:string

  Session-ID and name of the ILIAS client separated by :: e.g. 0f458b2cb7073821fa929a826dda97c6::ilias3

- active_id: xsd:long

  ID of the actual participant.

- question_id: xsd:long

  ID of the ILIAS test question.

- pass: xsd:int

  Number of test pass (starting with 0 for the first test pass).

- solution: tns:stringArray

  String array that contains the solutions of the participant and the scored points.

Description: Saves the results of a participant in the ILIAS database. solution is an array with N*3 elements that consist of triplets value1, value2, and points. Meaning: In value1, value2 you can send any kind of value you need

or your applet creates (values, text etc. because these elements are simple text based elements in the database). In points you typically will store the points of the partial solution sent in this triplet. All points will be summarized to the total points of the question and shall not exeeed the value in max_points. value1 and value2 are dependend on the structure of your question and help you to identifiy the solution of the participant. If you have for example a flash applet that provides a multiple choice question with single response, you could use value1 to store the index of the selected solution. value2 could be empty because there is nothing more you would like to store and points would be used to save the points for the selection. For a multiple choice question with multiple response you would define as many triplets as the participant selects answers in the question.

`saveQuestionSolution(sid, active_id, question_id, pass, solution)`

Parameter

- sid: xsd:string

  Session-ID and name of the ILIAS client separated by :: e.g. 0f458b2cb7073821fa929a826dda97c6::ilias3

- active_id: xsd:long

  ID of the actual participant.

- question_id: xsd:long

  ID of the ILIAS test question.

- pass: xsd:int

  Number of test pass (starting with 0 for the first test pass).

- solution: xsd:string

  XML-Code, der die Lösungen des Teilnehmers und die vergebenen Punkte beinhaltet.

Description: Saves the results of a participant in the ILIAS database in the same way as the saveQuestion function. The only diffence is that the solution isn't an array of triplets but an XML string of the following structure

```
<values>
 <value>VALUE</value>
 <value>VALUE</value>
 <points>POINTS</points>
 ...
</values>
```

The structure will be used in the same way as the solution triplets in the saveQuestion function.

`getNrOfQuestionsInPass(sid, active_id, pass)`

Parameter

- sid: xsd:string

  Session-ID and name of the ILIAS client separated by :: e.g. 0f458b2cb7073821fa929a826dda97c6::ilias3

- active_id: xsd:long

  ID of the actual participant.

- pass: xsd:int

  Number of test pass (starting with 0 for the first test pass).

Description: Returns the number of questions in the actual test pass.

`getPositionOfQuestion(sid, active_id, question_id, pass)`

Parameter

- sid: xsd:string

  Session-ID and name of the ILIAS client separated by :: e.g. 0f458b2cb7073821fa929a826dda97c6::ilias3

- active_id: xsd:long

  ID of the actual participant.

- question_id: xsd:long

  ID of the ILIAS test question.

- pass: xsd:int

  Number of test pass (starting with 0 for the first test pass).

Description: Returns the position of the question in the actual test pass.

`getPreviousReachedPoints(sid, active_id, question_id, pass)`

Parameter

- sid: xsd:string

  Session-ID and name of the ILIAS client separated by :: e.g. 0f458b2cb7073821fa929a826dda97c6::ilias3

- active_id: xsd:long

  ID of the actual participant.

- question_id: xsd:long

  ID of the ILIAS test question.

- pass: xsd:int

  Number of test pass (starting with 0 for the first test pass).

Description: Returns an array of floating point numbers with the resulting points of the questions in the actual test pass that occured before the question with the ID question_id.

`getQuestionSolution(sid, active_id, question_id, pass)`

Parameter

- sid: xsd:string

  Session-ID and name of the ILIAS client separated by :: e.g. 0f458b2cb7073821fa929a826dda97c6::ilias3

- active_id: xsd:long

  ID of the actual participant.

- question_id: xsd:long

  ID of the ILIAS test question.

- pass: xsd:int

  Number of test pass (starting with 0 for the first test pass).

Description: Returns the previous solution of a participant for this question. This function should be called when a question is opened to retrieve possible previous results of a participant and to present these results as a previous solution. If previous solutions are disabled in ILIAS this function will return nothing..

`getTestUserData(sid, active_id)`

Parameter

- sid: xsd:string

  Session-ID and name of the ILIAS client separated by :: e.g. 0f458b2cb7073821fa929a826dda97c6::ilias3

- active_id: xsd:long

  ID of the actual participant.

Description: Returns a string array existing of the user data of the actual test participant in the following format: "full name", "title", "firstname", "lastname", "login". For anonymous tests only "unknown" will be returned as full name. All other entries will be empty.